

CPFR Technical Specification

Electronic commerce is at a crossroads. For many buyers and sellers worldwide, Electronic Data Interchange (EDI) has become the backbone for computerized business-to-business communication. Meanwhile, the explosion of the Internet has brought universal access and a host of new technologies. The danger is that the benefits of electronic commerce standards such as EDI will be swept away by the excitement of the Internet, leading to a future world of proprietary platforms, incompatible APIs, and complex, unique collaborative trading practices among buyers and sellers.

CPFR intends to ensure that the industry actually captures the benefits of inter- and intra-enterprise collaboration through a common, pragmatic approach. Rather than create a new standard, it leverages the legacy of existing standards in broad use through the retailing and consumer manufacturing industries to develop guidelines for collaborative business processes (e.g., EDI and U.P.C). In this chapter, we explain the principles behind the CPFR technology specification, describe the data format standards, outline the transport/network protocol guidelines and security conditions, and present some application architecture considerations.

Principles

We envision CPFR as a platform- and vendor-independent environment where multiple parties can interoperate. Partners of different sizes and technical levels can collaborate through accessible technologies, including Internet and the Web, private Value Added Networks (VANs), dial-up, or other transport mechanisms. This communication is supported by formal standards, which evolve through an open process.

CPFR technological applications can take many forms. There are, however, several requirements to which they must adhere:



5.0 CPFR Technical Specification

- **Standards:** The system must use *existing* standards wherever possible. Where *de jure* standards have not been established, the committee has selected *de facto* standards that have an open process, that are managed by a non-profit organization, and are supported by multiple technology vendors. If these criteria have not been met in an area, the committee has declined to make a recommendation.
- **Scalability:** The system must be able to scale to large implementations in terms of number of products, trading partners, collaborative relationships, users, and collaboration interactions.
- **Security:** Data security is a major issue in a collaborative environment. For obvious reasons, sensitive information should be accessible only to those with permission to view it. CPFR technology solutions must ensure data is secure when exchanged via public networks, enabling robust collaboration without revealing data to competitors.
- **Open Design:** Solutions that require a single vendor's application are not acceptable in collaborative relationships that have no locus of control. There is no "master of the supply chain"; each trading partner must independently consider all of its buyer/seller relationships. It is unlikely that all of them would choose the same implementation approach. The technical specification must be robust enough to support the entire business process, yet simple enough to allow existing applications to connect with minimal adaptation. By using open and published standards, new trading partners can come online quickly and the systems can evolve. In addition, an open solution must be based on mature technologies, because the rapid pace of development and market acceptance can take evolving technologies in diverging directions, including extinction.
- **Manageability:** A collaborative solution must be easily maintainable by all parties. Custom solutions and annual software updates drive up costs, introduce incompatibilities, and cause downtime. Again, with no locus of control, a solution must be robust from the start, must not require inordinate support services, and must be able to survive over time. CPFR business practices should offer no technical or economic barriers for trading partners, large or small.

- **Resiliency:** The technical solution must be resilient to failure, not only in software, but also in the communications infrastructure. Redundancy provides greater reliability and increased capacity. In the event of hardware or power failure, the system should be fault-tolerant.
- **Collaboration:** Collaboration is more than messaging. With no central planning body, a CPFR solution must support threaded peer-to-peer communications among trading partners. It must also facilitate one-to-many communications among participants. The solution will almost certainly involve a combination of both human and machine input and output. There should be automated access to data from decision support software, execution systems, and so forth, as well as a facility for exception management by human operators.

Specification Approach

To derive the technical specification presented here, the CPFR technical subcommittee took the following approach. First, it analyzed the CPFR business process model and produced a set of data flow diagrams. From the data flow diagrams, the team then developed a logical data model and a data dictionary that illustrate the consolidated set of data elements and the relationships among them. These were then compared with the existing data format standards (ASC X12 and SIL) to identify an appropriate mapping and any gaps. The technical subcommittee concluded its work by creating guidelines and considerations for selection of transport, security, and application architecture mechanisms.

Technically, CPFR specifications, recommendations, and discussions of technical implementation criteria fall into four areas:

- **Data Format Standards:** Data formats for messages to be exchanged among CPFR trading partners, selected from the ANSI ASC X12 Electronic Data Interchange (EDI) standard, and the Standard Interchange Language (SIL) standard. CPFR data requirements will be included in the UCS and VICS EDI implementation guidelines, which are subsets of ASC X12, and the SIL guidelines.



5.0 CPFR Technical Specification

- **Transport/Network Protocol Guidelines:** Criteria for selecting the data transport (e.g., FTP) and underlying network protocol (e.g., TCP/IP) specifications for transmitting messages between CPFR trading partners.
- **Security Considerations:** Techniques for authentication, encryption, non-repudiation, and origin of CPFR messages that implementations should take into account.
- **Application/Middleware:** Alternatives for the location, coordination, and management of the data processing elements (servers, agents, and other components) that make up a CPFR implementation.

All CPFR implementations must use the data formats described in this specification for message interchange. The selection of data transport, security scheme, and middleware is beyond the scope of the CPFR standard, however, and is subject to implementers' agreements. The guidelines and selection criteria provided here should help trading partners agree on which approach to use.

Conceptual Model

The CPFR conceptual model has been divided into a set of process flow models, data flows, a logical data model, and a data dictionary. The CPFR process flow models are described in the "Future Process State" chapter. This section describes the other models.

Data Flows

Each CPFR trading partner interaction produces a data flow, which is translated to standard message formats and data transport requests in online implementations.

Table 3 summarizes each of the CPFR data flows.

Data Flow Summary

Process Step	Data Consumed	Data Produced
Develop Front End Agreement	(None; manual process)	(None; manual process)
Create Joint Business Plan	Buyer's Corporate Strategy Seller's Corporate Strategy	Joint Business Plan
Create Sales Forecast	Joint Business Plan POS Data Event Sales Forecast Revisions	Sales Forecast
Identify Sales Forecast Exceptions	Sales Forecast Exception Criteria Metrics Events	Identified Exception Items
Collaborate on Sales Forecast Exceptions	Buyer's Secondary Data for Exception Items Identified Exception Items Seller's Secondary Data for Exception Items	Sales Forecast Item Revisions
Create Order Forecast	Order Forecast Revisions POS Data Current Inventory on Hand Inventory Strategy/Seasonal Info Sales Forecast Events Product Historical Demand and Shipments Product Availability Data Item Management Profile Data	Order Forecast
Identify Order Forecast Exceptions	Order Forecast Exception Criteria and Values Events	Identified Order Exception Items
Collaborate on Order Forecast Exceptions	Buyer's Secondary Data for Exception Items Identified Exception Items Seller's Secondary Data for Exception Items	Order Forecast Revisions
Generate Order	Order Forecast Item Management Profile	Order

Table 3

The following pages illustrate the high level processes and the data flows associated with these processes. These data flows are independent of the architectural implementation of CPFR. In other words, there could be different architectural implementations of this model, such as peer-to-peer, hub-and-spoke, and third party. For example, the CPFR data could be managed in a domain shared (or potentially duplicated) by the buyer and the seller, or it may exist within one of the two partner domains, or it could even exist within a third-party service. The important point about the illustration is the agreement on the types and formats of the data being shared and the nature of the data flows.



5.0 CPFR Technical Specification

Note: When an individual diagram includes multiple process steps, the steps are numbered using the step numbers in the CPFR IDEF0 diagrams.

Develop Front End Agreement

The *Develop Front End Agreement* data flow illustrates the process of coming to agreement on joint business goals for the trading partner relationship.

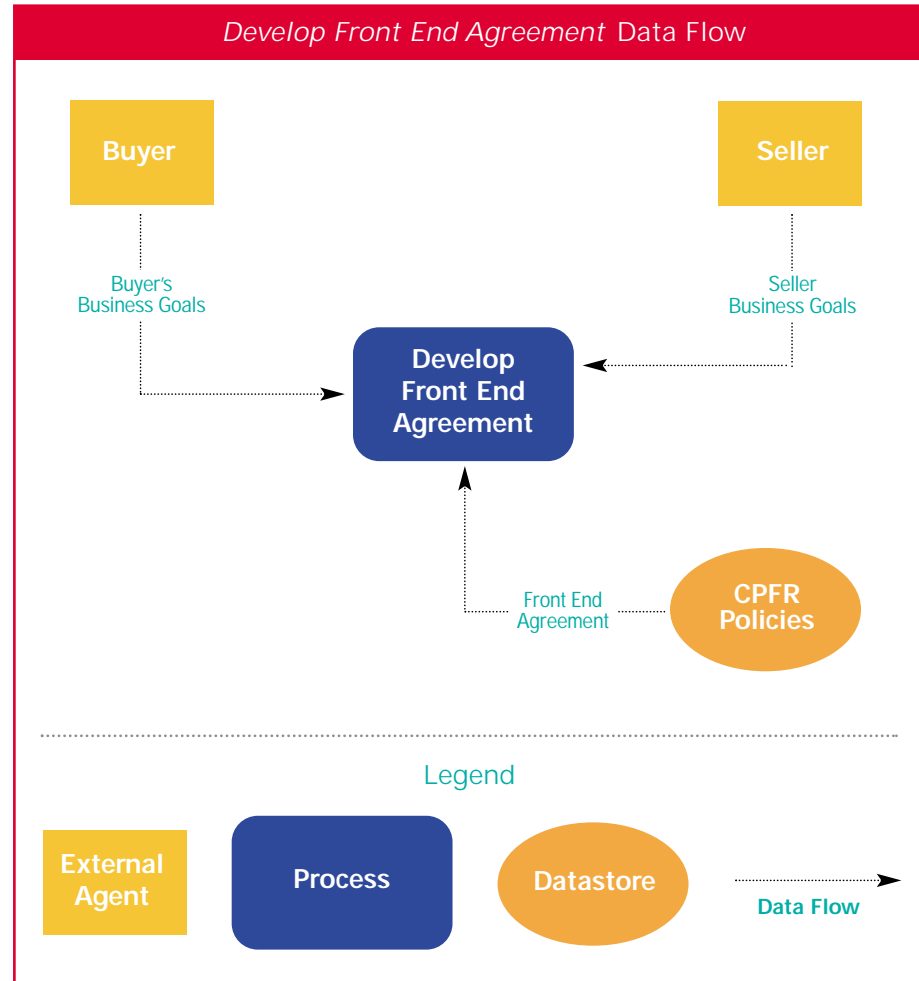


Figure 16

Create Joint Business Plan

The *Create Joint Business Plan* data flow covers the exchange of strategy, objective, and goal setting information between CPFR trading partners at the beginning of a planning period.

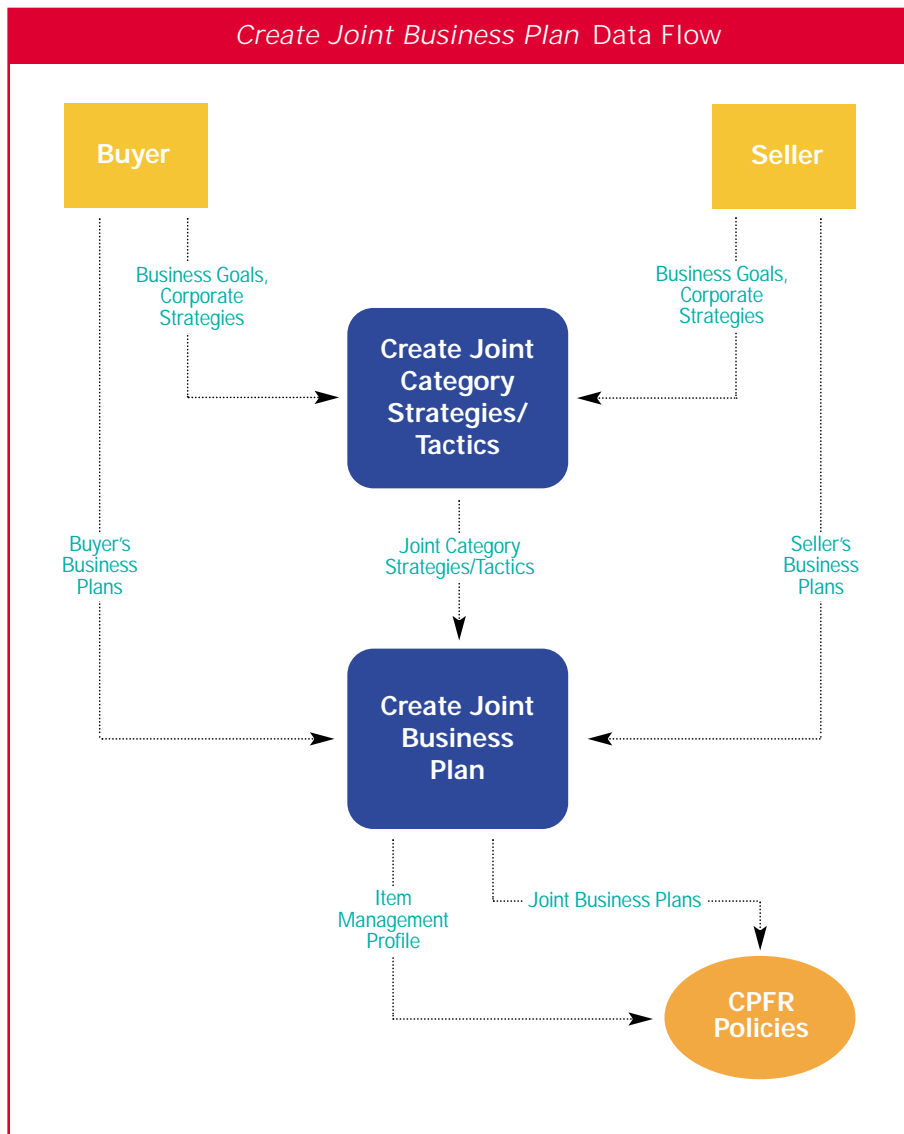


Figure 17



5.0 CPFR Technical Specification

Create Sales Forecast

The *Create Sales Forecast* data flow describes the exchange of initial sales forecasts for a planning period, based on agreed-upon goals, events, sales results, and prior revisions.

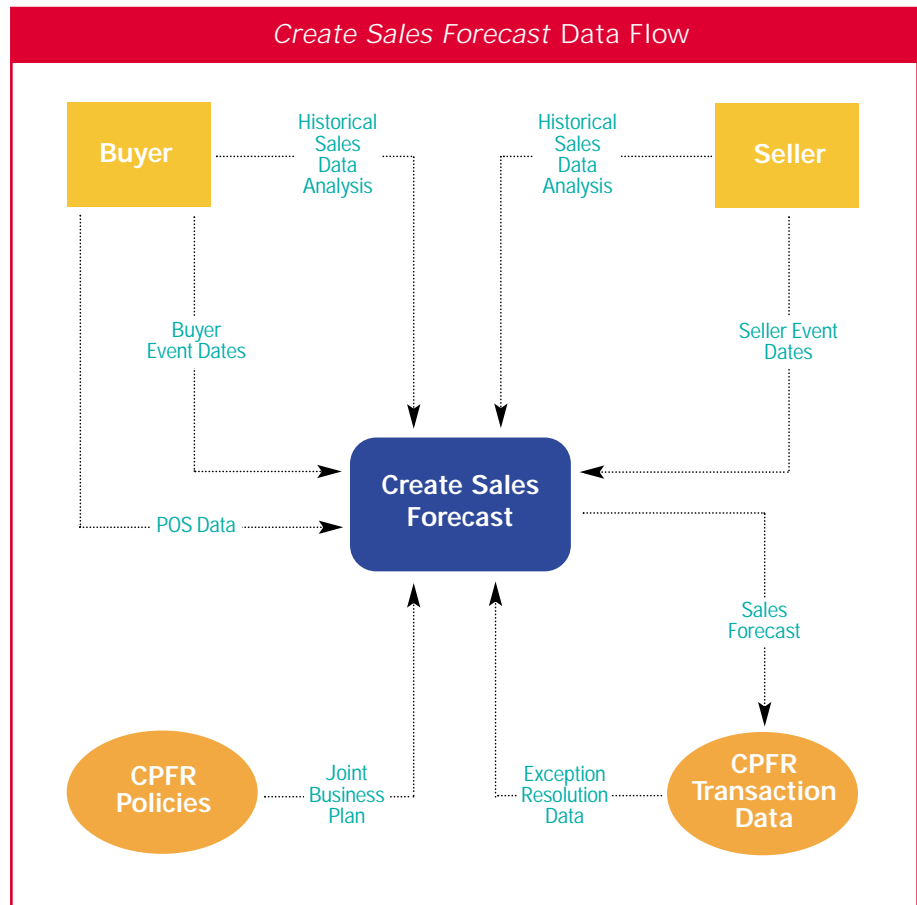


Figure 18

Identify Sales Forecast Exceptions

The *Identify Sales Forecast Exceptions* data flow documents the creation and exchange of exception items related to forecast performance that result from criteria established in the joint business plan.

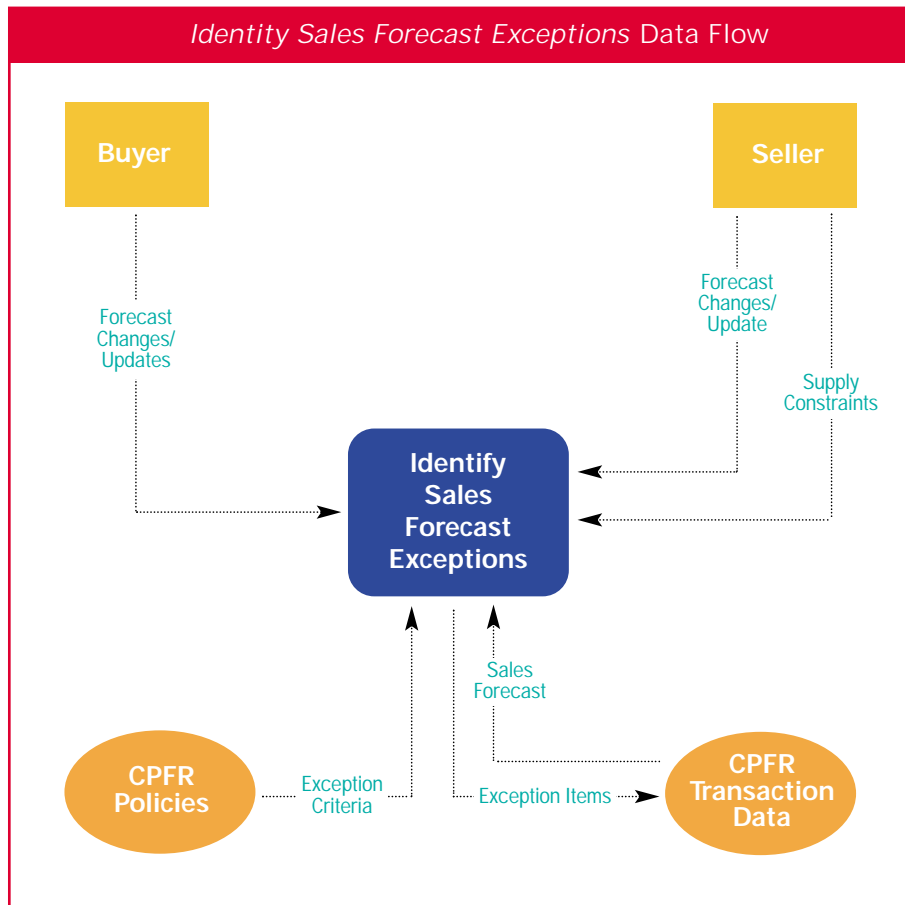


Figure 19



5.0 CPFR Technical Specification

Resolve Sales Forecast Exceptions

The *Resolve Sales Forecast Exceptions* data flow captures the exchange of sales forecast item revisions created to resolve forecast exceptions.

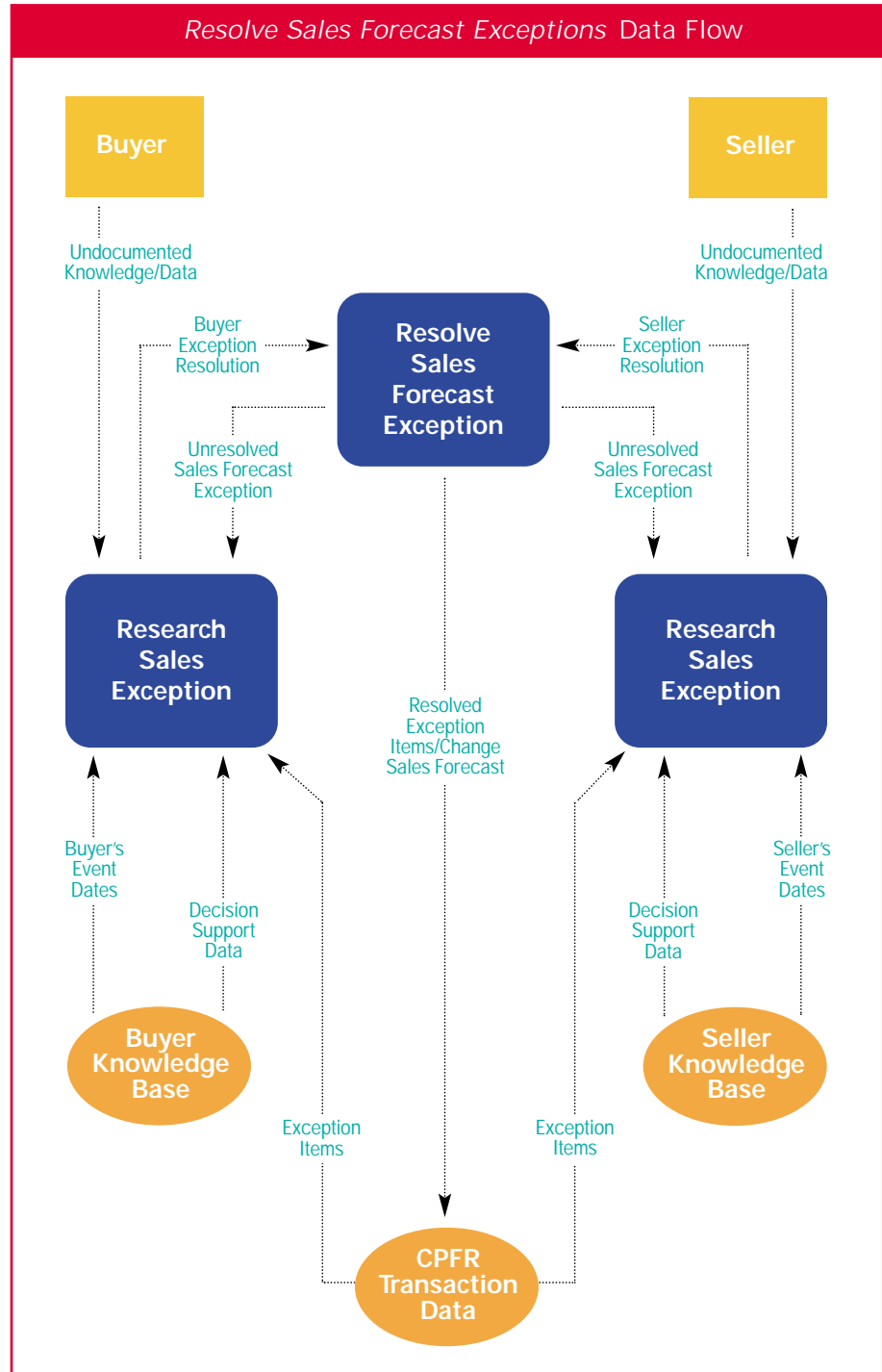


Figure 20

Create Order Forecast

The *Create Order Forecast* data flow describes the information exchanged in an initial order forecast for products within a planning period.

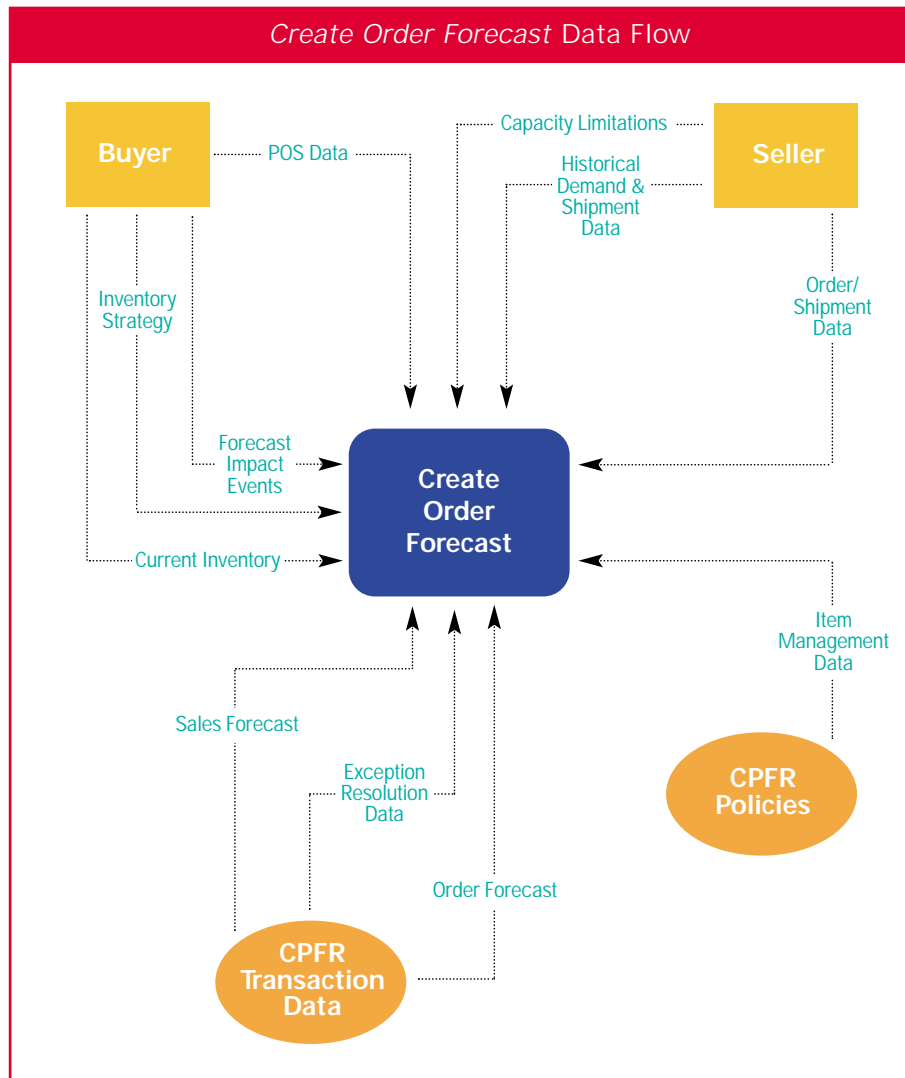


Figure 21



5.0 CPFR Technical Specification

Identify Order Forecast Exceptions

The *Identify Order Forecast Exceptions* data flow illustrates the information exchanged when an order forecast triggers exceptions based upon joint business plan criteria.

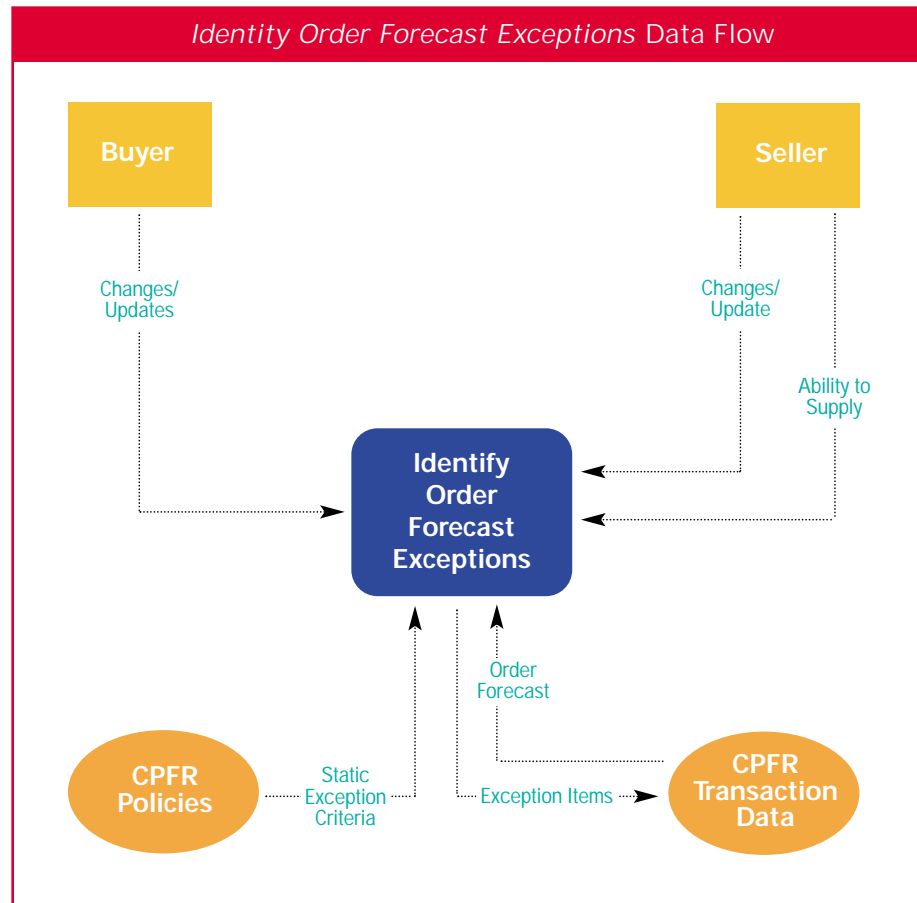


Figure 22

Resolve Order Forecast Exceptions

The *Collaborate on Order Forecast Exceptions* data flow captures the information exchanged when CPFR trading partners create revisions to order forecasts in order to resolve an exception condition.

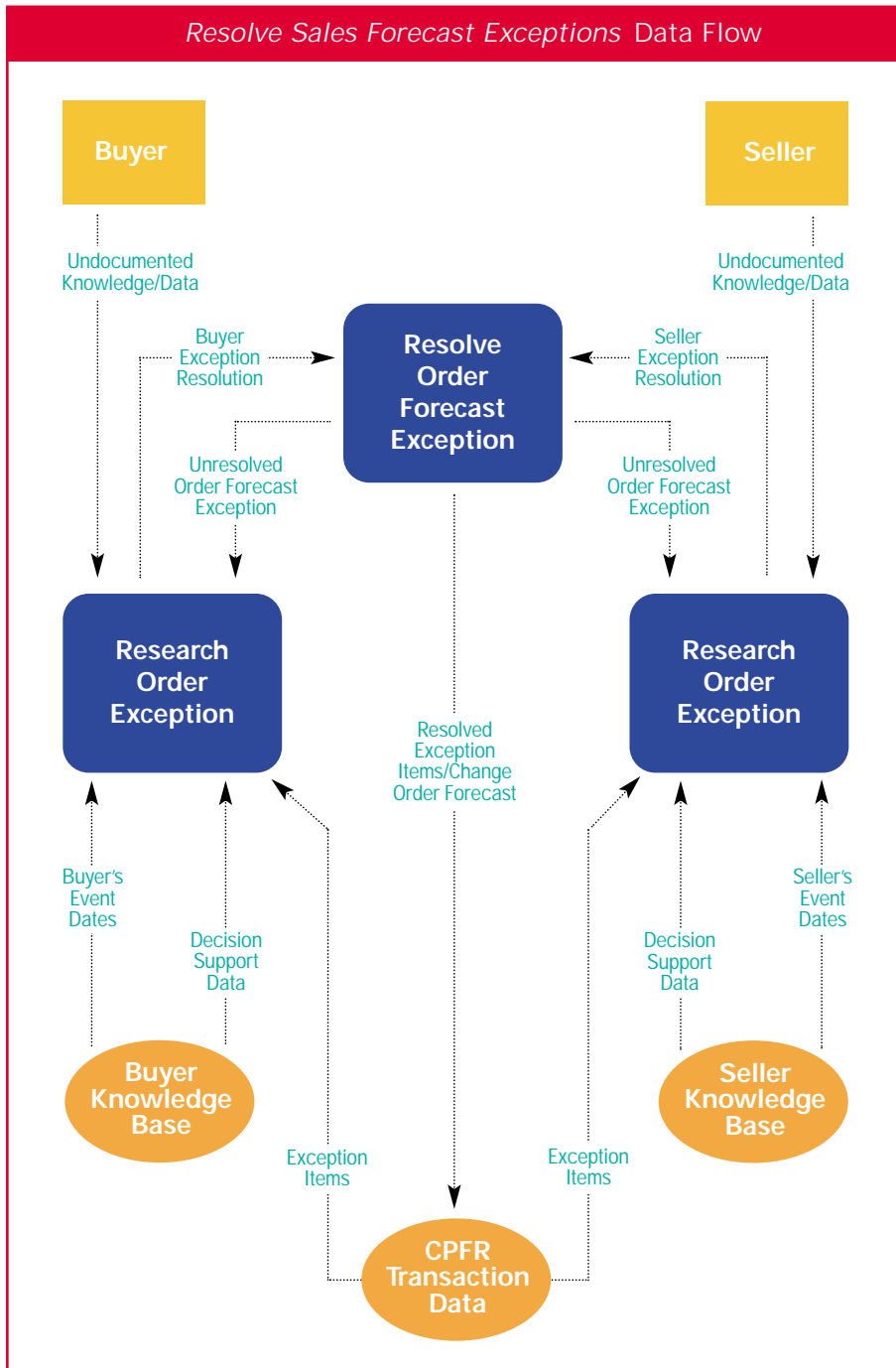


Figure 23



5.0 CPFR Technical Specification

Generate Order

The *Generate Order* data flow documents the transmission of a firm order for products, based upon an order forecast and an item management profile.

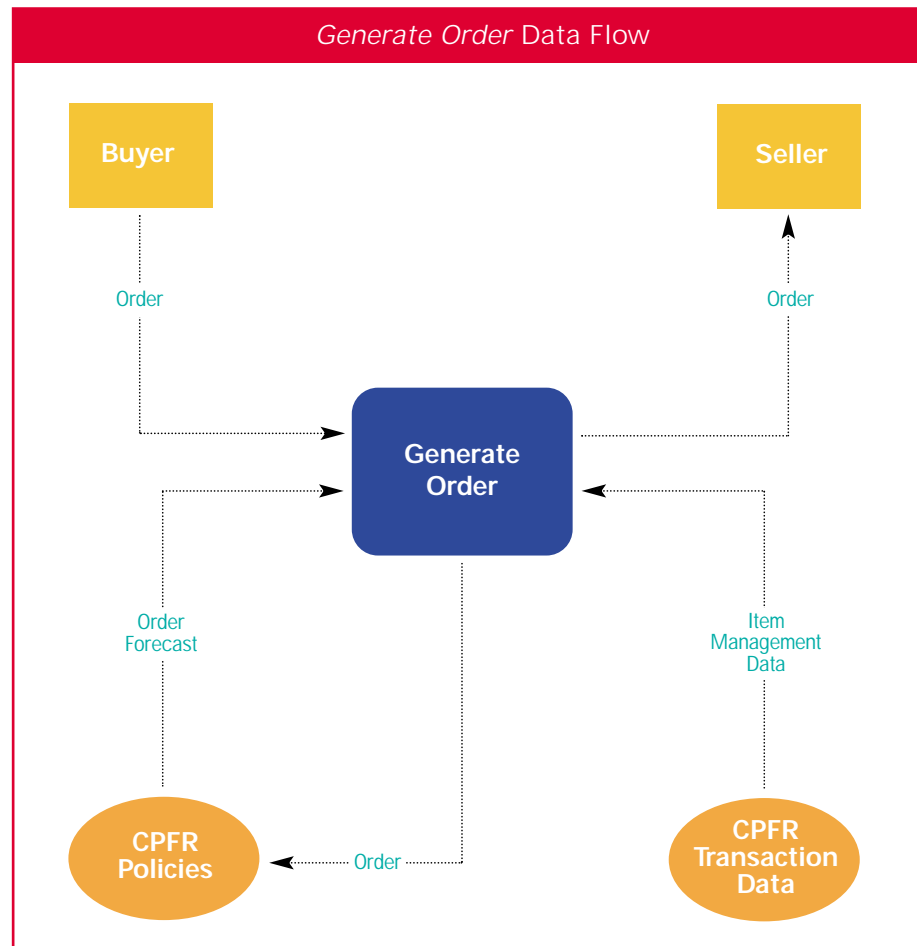


Figure 24

Logical Data Model

The consolidated set of data elements required to produce the CPFR data flows and the logical relationships among them are included in the CPFR logical data model. The data model is in relational form (using the IDEF 1X relational modeling standard) to aid in the construction of Standard Interchange Language (SIL) data format standard requests (discussed in the Data Format Standards section of this chapter). Appendix C presents the entity-relationship diagram for the model.

This model is provided as a reference; CPFR implementations are not required or expected to include a physical database with this schema. Not all fields will be used by each trading partner.

Data Dictionary

The CPFR data dictionary explains the meaning and intent of each logical data element referenced in the CPFR data model. A common data dictionary guides the mapping of CPFR data elements to data format standards such as ANSI ASC X12 EDI and Standard Interchange Language (SIL). Appendix D contains a complete listing of CPFR data dictionary terms.

Data Format Standards

Selection of Standards

Data format standards specify the order, types, and size of data to be exchanged in files or messages. They do not specify how the data is transmitted or secured. Standardizing the data format of CPFR messages ensures that implementations can interpret communications received from CPFR trading partners. Each CPFR message is specified in one of two data format standards: ANSI ASC X12 EDI or Standard Interchange Language (SIL).

Use of ANSI ASC X12 Transaction Sets for CPFR Messages

CPFR uses ANSI ASC X12 transaction sets to exchange messages. To support the *Create Sales Forecast* data flow, the Planning Schedule with Release Capability [830] transaction set is used to transmit the sales forecast among trading partners. To support the Actual Item Performance History Request message, the Item Information Request [893] transaction set is used to request information for specific products, locations, and time periods. The Product Activity Data [852] transaction set supports the Actual Item Performance History Response message and is used to transfer product activity or metrics among trading partners. This data can be compared with the forecasts exchanged to determine forecast accuracy. The Purchase Order [850] or the Grocery Product Purchase Order [875] transaction set transfers the order during the *Generate Order* data flow



5.0 CPFR Technical Specification

scenario. The Price/Sales Catalog [832] or the Item Maintenance [888] transaction set is used to update product information to trading partners. The Promotion Announcement [889] transaction set is used to inform CPFR trading partners of upcoming promotional events. Appendix E maps CPFR requirements to these transaction sets. Other EDI transaction sets provide equivalent data, which could be used to implement CPFR. However, this specification does not provide any mapping of them. (See the VICS EDI and UCS implementation guidelines for maps.)

EDI standards as they currently exist do not provide transactions for some of the collaboration scenarios represented in CPFR. For example, the establishment of joint product goals for out-of-stock and inventory levels, exception conditions, and a rationale for revisions made are all outside the scope of EDI. The CPFR specification provides a mapping to Standard Interchange Language (SIL) for these message types.

Use of SIL for CPFR Messages

The Standard Interchange Language (SIL) standard, maintained by the Uniform Code Council (UCC), is a data interchange language based on ANSI Structured Query Language (SQL) syntax. A separately maintained data dictionary defines the standard table and field names that may be exchanged in SIL messages. SIL benefits include the familiarity of SQL syntax, the self-describing nature of its messages, and the ability to both update and query remote systems.

The SIL standard is not nearly as widely known or used as EDI. Since its introduction in 1990, SIL has been used to respond to dynamic data interchange challenges among grocers and grocery distributors (for example, distributing price changes among disparate point-of-sale systems or querying stock positions).

The CPFR committee has chosen to use the SIL approach during the pilot phase to document messages that are not within the scope of EDI today. SIL's data dictionary committee uses a streamlined approach to approve enhancements to the standard very rapidly, providing a convenient forum for rapid standardization. SIL is the only forum that the committee is aware of that would allow CPFR to

achieve standards body coverage for its entire specification, without requiring the CPFR committee to maintain the standard as it evolves. When the appropriate message types are added to EDI standards they will be considered for use in CPFR.

The Extensible Markup Language (XML) is another data formatting standard that the CPFR committee is considering for future versions of the CPFR guidelines. Use of XML is contingent upon the presence of a standards organization that can maintain the XML Document Type Definition (DTD) that the CPFR committee would produce. To date, no appropriate organization is in place, but progress in this area appears imminent.

Appendix E specifies the mapping of CPFR message requirements to existing SIL data dictionary elements and identifies required elements for future SIL extensions.

Transport/Network Protocol Guidelines

CPFR implementations must agree on the transport and network protocol to be used. TCP/IP has become the *de facto* standard protocol for public networks; private value-added networks also widely deploy IP. Other networks are technically capable of supporting CPFR, including standard protocol stacks such as ISO OSI. Since most contemporary computing architectures can support multiple protocols simultaneously, and virtually all support TCP/IP, network selection is not likely to be a major implementation concern.

The appropriate physical medium for transporting data depends on the architecture selected and the level of service desired. CPFR-formatted messages may be exchanged over the selected network in files, as data streams, or as blocks of data delivered through a messaging system.

HTTP/S

The Hypertext Transport Protocol (HTTP) and its secure variant (HTTP/S) are the transport specifications used by Web browsers to transmit Hypertext Markup Language (HTML), Java applets, images, and other content to and from servers.



5.0 CPFR Technical Specification

HTTP was not originally designed to transport highly structured data or to manage complex interactions of clients and servers. However, its pivotal role in the growth of the Internet has propelled its evolution. A number of EDI software companies now offer Internet-enabled EDI, which allows EDI messages to be exchanged over HTTP.

The advantage of HTTP is its wide acceptance. Most organizations allow HTTP messages through Internet security firewalls. HTTP can also be used synchronously, enabling interactive response.

FTP

The File Transfer Protocol (FTP) is a widely supported means of transferring text and binary files among heterogeneous systems. Properly authorized users from other organizations may transfer files to or from an FTP server and successfully interpret their contents. Files can be easily organized and archived, and failed transmissions can be easily reinitiated without data loss. It is more difficult to support interactive implementations with a file-based approach, however. Typically file exchange occurs as a background activity.

SMTP and MIME

E-mail systems and their related standards are another option. Simple Mail Transfer Protocol (SMTP) is the *de facto* Internet standard for e-mail transmission. While many e-mail systems support SMTP today, not all of them are interoperable. Multi-purpose Internet Mail Extension (MIME) provides structured data types for SMTP that could be used to format CPFR messages. Efforts are underway to provide a MIME extension for CPFR.

Other Approaches

Sockets

Sockets are a data streaming protocol. More flexible and primitive than other approaches, sockets require tight agreement among implementations concerning the control flow, recovery, segmenting, and other strategies to be used. Sockets are available on all computing platforms. They can provide high performance; however, more programming is involved.

Message-oriented Middleware

There is a relatively mature market of message-oriented middleware that provides data transmission, security, assured delivery, multicasting, broadcasting, publish/subscribe messaging, and other advanced options. Standards for messaging products are extremely primitive. For this reason, implementations using message-oriented middleware usually have to agree on a vendor to use in common.

OMG CORBA IIOP

The OMG CORBA specification provides a protocol called IIOP (Internet Inter-ORB Protocol) for making remote object requests and collecting results. CPFR is currently defined in terms of data to be transmitted, rather than object interactions. As a result, IIOP requires implementations to take the extra step of mapping CPFR messages to objects. Future revisions of the CPFR specification may introduce a distributed object model that could be used more effectively with this approach.

Security Considerations

Selection of a specific security standard is outside the scope of the CPFR technical specification. However, every implementation will need to consider how communication will be secured, especially when messages are exchanged over public networks. This section provides background information on security issues that may be useful for those organizations contemplating CPFR initiatives.

Key attributes that should be accessed in a system that provides and controls system access include authentication, authorization, integrity, confidentiality, auditing, and non-repudiation.

- **Authentication:** The security system needs to authenticate remote users, systems, and applets or other downloaded code. This means ensuring that users connecting to the system are who or what they are supposed to be, and ensuring that the remote systems to which users are connected are the ones they meant to address and not impostors. There also needs to be a means to ensure what is being downloaded is the correct code and that a substitution has not taken place. Some standards have been established in this area. For example, there is certificate code that supports X.509v3 and provides an online Certificate Authority server to provide user authentication.



5.0 CPFR Technical Specification

- **Authorization:** Assuming a user is “authentic,” the next area a security system needs to address is the authorization to a system or application and the level of access granted. This includes application privileges for field or record level access, write permissions, and the ability to assign rights to other users. With numerous users working at many different locations, an effective and efficient process to administer users at a group level needs to be incorporated into the security system. Ideally, each group would have an administrator who would grant lesser levels of access and authorization to the other members. By restricting the system access to a user and restricting the scope of that access (database views vs. direct table access, for instance), an administrator can minimize the potential damage from a system breach. Most controls already in place rely on individual applications for containment. Additional measures must be instituted to provide containment for a specific system or application server(s).
- **Integrity:** A digital signature attests to the contents of a message as well as to the identity of the signer. As long as a secure hash function is used, there is no way to take someone’s signature from one document and attach it to another or to alter a signed message in any way. The slightest change in a signed document will cause the digital signature verification process to fail. Thus, public-key authentication allows people to check the integrity of signed documents. If a signature verification fails, however, it will generally be difficult to determine whether there was an attempted forgery or simply a transmission error.
- **Confidentiality:** At a system level, confidentiality means that all attributes of a connection, including time, frequency, data, queries, and so on, remain confidential. For CPFR, the primary focus is on the data being accessed, received, and sent, regardless of the means used. At present, collaboration is largely by phone and fax. The security with this approach is almost nonexistent. The amount of data transmitted electronically is very small or sent over secure private networks. As real-time access to data or transactions becomes a business requirement and the medium becomes the Internet, the requirements for confidentiality are best met through some form of encryption.

- **Auditing:** The security system must provide for the auditing of access and potential breaches. The audit trail provides a mechanism for assessing the extent of possible damage and the nature of the attack, and for assisting in the development of protection against future attacks. The system can also provide a means for alerting administrators to attempted and actual breaches of the existing security systems.
- **Non-repudiation:** Non-repudiation is key to ensuring that outside parties involved in initiating transactions within the applications are tied to those transactions. Authentication at the appropriate level of granularity ensures that users of the system cannot deny knowledge of, or responsibility for, a transaction. This is critical to generating business transactions using these systems in place of traditional paper-based processes.

Application Architecture Considerations

The remainder of this chapter explains technical scenarios under which CPFR could be deployed. Every network of CPFR partners will need to agree on the location, coordination, and management of data-processing elements in their implementation. Depending on each case scenario, the resources available, and the need for expediency, the architecture may follow one of the application architectures described in this document.

Implementation Scenarios

CPFR implementations can range from real-time, Internet-based applications to file-based exchanges in batch over a VAN. E-mail and FTP (file transfer protocols) are also viable. Data could be managed by each trading partner or by an independent, third-party service bureau. Selection of the transport mechanism depends on the level of service required, the availability of the technology to each trading partner, and the ease of deployment. In all cases, the message content exchanged over these transports should conform to the CPFR selection of data format standards.



5.0 CPFR Technical Specification

Some aspects of collaboration are time-critical. The level of sensitivity to response times varies dramatically, depending on the products and trading partners involved. The CPFR technical guidelines do not include formal specification of any level-of-service requirements other than to identify which data may be time-critical in the data flow model. Implementations must negotiate level-of-service agreements and select transports, and design their applications to achieve appropriate response times in each communication context.

There are many alternatives for distributing data and information processing among collaborating trading partners. Each of these collaboration environments presents significant tradeoffs. Examples of each are likely to prevail in different industries, depending on the rate of market acceptance of CPFR.

Coexistence

Coexistence mirrors most electronic commerce schemes today. Each enterprise operates its own autonomous network of applications and exchanges business documents with its trading partners as a background activity, processing the documents nightly or weekly. Figure 25 illustrates this type of trading partner network. This style is peer-to-peer, but uncoordinated.

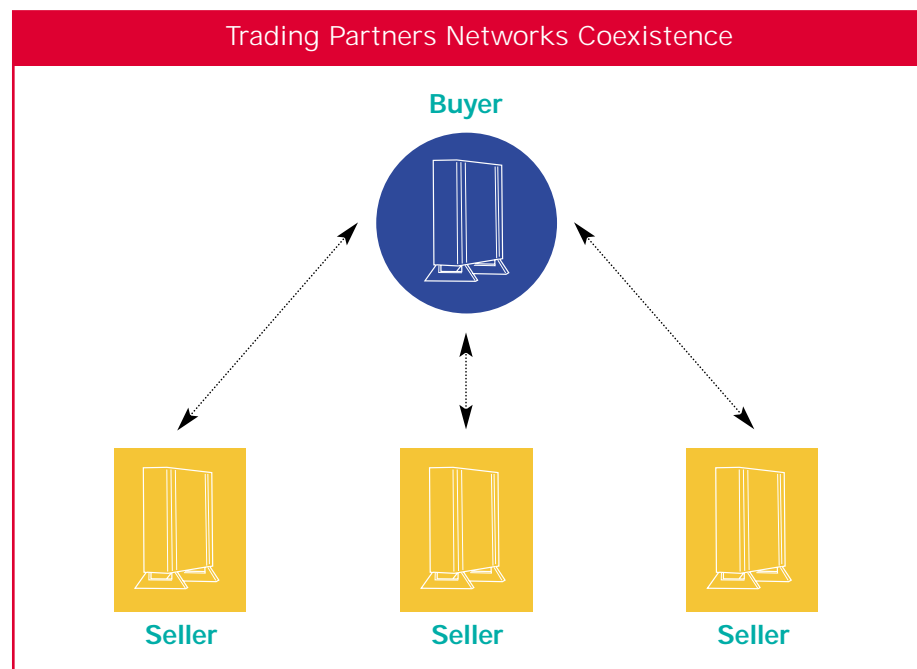


Figure 25

The advantage of these networks is that they are mature, widely deployed, and frequently supported by the ANSI ASC X12 EDI message types required by the CPFR specification. However, because of the focus on batch execution, an environment based solely on coexistence is not well suited to time-critical collaborative exchanges.

Centralized Server

In supply chains that have a few large and many small players, one company can develop and manage a collaboration environment on behalf of its trading partners. Because a significant part of any of the smaller companies' production flows through the larger trading partner, and the burden is not on them to develop or maintain the system, smaller companies are usually motivated to adopt their partner's technology. Figure 26 shows a centralized environment, in which the server is managed by a buyer and clients are operated by sellers.

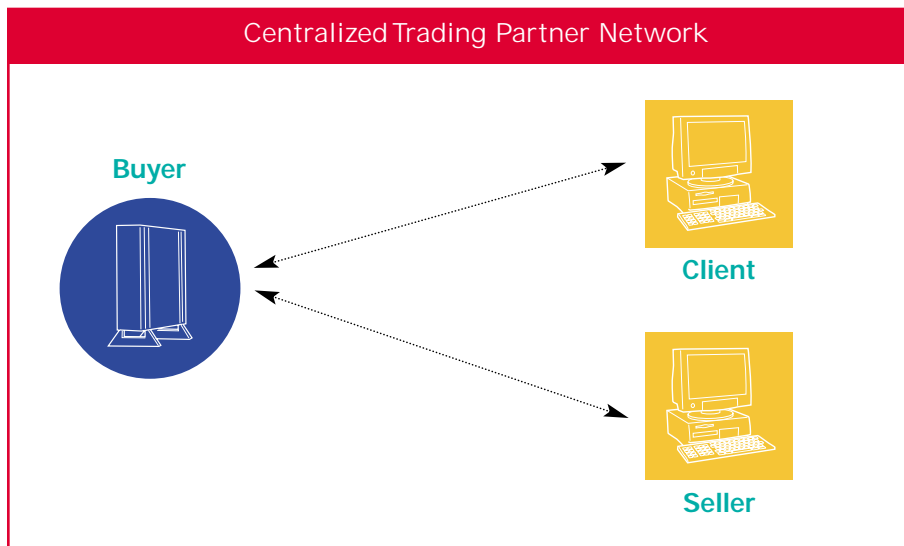


Figure 26

Distributed Servers

A system built on a network of distributed servers can locally manage the data relevant to any individual company and restrict distribution of sensitive information to trading partners that need it. Distributed server networks are highly scalable because the number of systems grows with the number of trading partners.

Figure 27 illustrates a distributed server architecture.



5.0 CPFR Technical Specification

Distributed server networks are more complex than centralized or loosely coupled systems, so they require more sophisticated engineering and management. The distributed server approach also requires each trading partner to install and maintain its own system, which could be an impediment to the widespread adoption of CPFR. Data synchronization is the most challenging implementation issue, whether performed via database replication, distributed transactions, or other means.

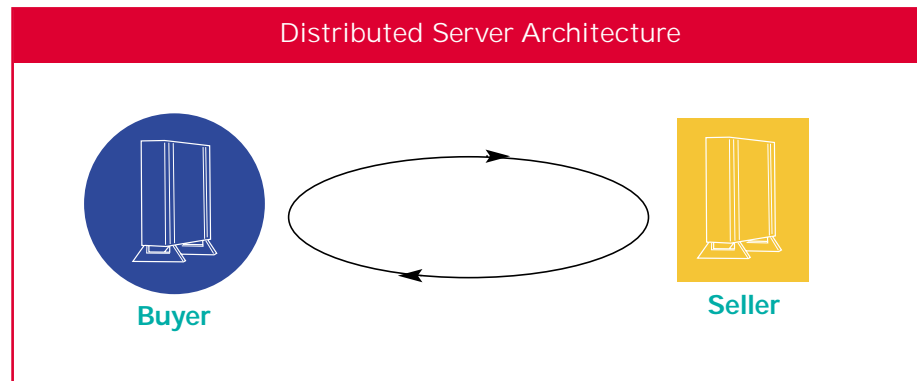


Figure 27

Summary of Systems Architecture Options for CPFR

Architecture	Description	Strengths	Weaknesses
Coexistence	Uncoordinated batch data transfer among trading partners	Minimal investment required	Reaction time may be too slow for true collaboration
Centralized Server	All data for a company and its trading partner network maintained in its own server(s)	<ul style="list-style-type: none"> Centralized management Ease of entry for partners 	Partners lack visibility control of data across other buyer/seller in relationships
Distributed Servers	Peer-to-peer network of servers installed at each trading partner	Each partner controls and has visibility of all its own data	Higher investment and IT capability required

Table 4

Emerging Technologies

Over the past 10 years, a growing proportion of application programs has been written using object-oriented techniques. Objects combine the data and behavior of a business entity—such as a forecast or an order—into a single unit. Distributed object environments allow individual objects to be accessed remotely as though they were local.

Sellers and buyers have not deployed systems based on distributed objects in any significant capacity to date. Competing specifications, relatively immature products, and a lack of object-oriented programming skills on corporate IT staffs have limited adoption. Many of these limitations are likely to disappear soon, so it merits considering how to evolve CPFR to a distributed object specification in the future.

Distributed object approaches have a number of advantages over traditional application protocol specification techniques. Special variants of a specification can be created without modifying the base system through a process called sub-typing. Object-based protocols are not restricted to data flows; actions that are allowed to be performed on business entities can be explicitly specified through object methods. Finally, data transport and formatting concerns are simplified because distributed object specifications take care of these details.

The Object Management Group's (OMG) Common Object Request Broker Architecture (CORBA) is a public specification for distributed objects that has been used by other standards groups. It has been available since 1991. Many technology companies offer implementations of CORBA that can be used to construct distributed object environments. A popular object-oriented programming language, Java, provides its own distributed object technique called Remote Method Invocation (RMI), though it supports CORBA as well. Finally, Microsoft has a distributed object specification called DCOM, which it submitted to The Open Group to be maintained as though it were a standard.



5.0 CPFR Technical Specification

Conclusion

The CPFR technical specification is a work in progress. The CPFR technical subcommittee has selected the combination of standards and conventions that best meets the needs of supply-chain trading partners, based on the technologies available. Technologies in this domain continue to evolve rapidly. To validate the CPFR business process and continue to develop the technical recommendations for CPFR, the subcommittee encourages members to conduct reference pilots. Through the piloting process, the technologies outlined in this document will be tested in the context of real business situations and under field conditions. The committee then hopes to publish revisions to this document that will enhance these guidelines and standards selections.